

REMARKS

Applicant respectfully requests reconsideration and allowance of the subject application in view of the foregoing amendments and the following remarks.

Rejections under 35 U.S.C. §101

The Office rejected Claims 6 and 33 under 35 U.S.C. §101 because the claimed invention is allegedly directed to non-statutory subject matter. Applicant respectfully traverses this rejection.

Nevertheless, without conceding the propriety of the rejection and in the interest of expediting allowance of the application, Applicant has amended Claim 6 to recite “granting permissions on a computer system based on the permission grant set.” Applicant has also amended Claim 33 to recite “wherein permissions are granted on a computer system based on the permission grant set.” Granting permissions on a computer system is a tangible result and has practical application. Accordingly, Claims 6 and 33 are directed to statutory subject matter under 35 U.S.C. §101.

Rejections under 35 U.S.C. §102

The Office rejected Claims 1, 2, 4-11, 13-18, 20-27, 29-33, and 41-48 under 35 U.S.C. §102(e) as allegedly being anticipated by U.S. Patent No. 6,044,467 to Gong (“Gong”). Applicant respectfully traverses this rejection.

Claims 1, 6, 17, 22, 33, and 44 as amended recite:

1. A method comprising:
receiving a manifest defining a plurality of code assemblies that are members of at least one application, wherein the manifest defines at least one trusted application and application evidence for making a trust decision;
evaluating the application evidence to determine if the at least one application is trusted;
generating a permission grant set for each code assembly that is a member of the at least one application if the application evidence satisfies at least one condition for trusting the at least one application; and
passing the permission grant to a run-time call stack.
(Emphasis added).

6. A method comprising:
generating a permission grant set for each code assembly that is a member of at least one application if application evidence for the at least one application satisfies at least one condition specified in a security policy specification for trusting the at least one application, wherein the security policy specification defines multiple policy levels; and
granting permissions on a computer system based on the permission grant set.

17. A computer program product encoding a computer program for executing on a computer system a computer process, the computer process comprising:
receiving a manifest defining a plurality of code assemblies that are members of at least one application, wherein the manifest defines at least one trusted application and application evidence for making a trust decision;
evaluating the application evidence to determine if the at least one application is trusted; and
generating a permission grant set for each code assembly that is a member of the at least one application if the application evidence satisfies at least one condition for trusting the at least one application. (Emphasis added).

22. A computer program product encoding a computer

program for executing on a computer system a computer process, the computer process generating a permission grant set for each code assembly that is a member of at least one application if application evidence for the at least one application satisfies at least one condition specified in a security policy specification for trusting the at least one application, wherein the security policy specification defines multiple policy levels. (Emphasis added).

33. A system comprising:
a manifest defining at least one application;
application evidence to determine whether the at least one application is trusted; and
a policy manager to evaluate the application evidence relative to at least one condition, wherein the policy manager generates a permission grant set for each code assembly that is a member of the at least one application if the application evidence satisfies the at least one condition specified in a security policy specification for trusting the at least one application, wherein the security policy specification defines multiple policy levels, and wherein permissions are granted on a computer system based on the permission grant set. (Emphasis added).

44. A computer-readable medium having stored thereon a data structure, comprising:
a first data field specifying members of at least one application;
a second data field containing application evidence to evaluate whether the at least one application is trusted, wherein permission grant sets are generated for each member of the at least one application if the application evidence satisfies at least one condition specified in a security policy specification for trusting the at least one application, wherein the security policy specification defines multiple policy levels. (Emphasis added).

Gong U.S. 6,044,467

Gong discloses a method of resolving ambiguous class names when an object class is loaded into a computer system. (Column 9, lines 23-26). When

object classes (data indentifying a class) must be loaded into the computer system, a search is performed for code associated with the class. The locations (sources of code) are searched in a predetermined sequence that ensures that untrusted code will not be executed when trusted code with the same class name is available. If during the search the class is found and has not been loaded, the class is loaded and a set of permissions associated with the class based on a predetermined mapping of sources of code to permissions. (Column 6, lines 30-43).

Regarding Claims 1 and 17

Gong does not disclose “receiving a manifest defining a plurality of code assemblies that are members of at least one application, *wherein the manifest defines at least one trusted application and application evidence for making a trust decision,*” as recited in Claims 1 and 17. (Emphasis added)

Gong discloses that “an object is a record of data combined with procedures and functions that manipulate the record. All objects belong to a class. Each object belonging to a class has the same fields and the same methods. The methods are procedures, functions, or routines used to manipulate the object. (Column 7, lines 20 – 33). In other words, an “object class” is a collection of objects (*i.e.*, data and functions) with the same fields and methods.

Accordingly, Gong fails to disclose a “manifest defining *at least one trusted application and application evidence for making a trust decision,*” as recited in Claims 1 and 17. (Emphasis added)

Regarding Claims 1, 17, 33, and 44

Gong also does not disclose “*evaluating the application evidence to determine if the at least one application is trusted,*” as recited in Claims 1 and 17, and the similar features disclosed in Claims 33, and 44. (Emphasis added)

Gong discloses that “when an object class is loaded in to a computer system, a search is performed for the code [name] associated with the class ... if during the search, the class [name] is found and the class has not been loaded, the class is loaded and a set of permissions is associated with the class ...” (Column 6, lines 30-43). Contrary to the Offices’ assertion, Gong simply searches system memory for code that is associated with a class name, and if the code is present, loads the trusted code. Gong does not *evaluate evidence*, nor does it *determine whether the application is trusted*. Gong simply searches for code and loads that code if it is present in system memory.

Accordingly, Gong fails to disclose “*evaluating the application evidence to determine if the at least one application is trusted,*” as recited in Claims 1 and 17, and the similar features disclosed in Claims 33 and 44. (Emphasis added)

Regarding Claims 6, 22, 33, and 44

Gong fails to disclose “generating a permission grant set ... *if application evidence for the at least one application satisfies at least one condition specified in a security policy specification for trusting the application, wherein the security policy specification defines multiple policy levels,*” as recited in Claims 6, 22, and 33, and the similar features recited in Claim 44. (Emphasis added).

Gong discloses that “if the code sources in the predetermined mapping match the code source of the class ... then the class is assigned all or some of the permissions mapped to the code sources that match the code source of the class. (Column 6, lines 45-50). Gong further discloses that “code source[s] represent a source of code from which code is received, such as a particular set of one or more files or code stream from a trusted source or untrusted source.” (Column 7, lines 53-56). Accordingly, Gong simply compares the “mapping match” code source with the “class” code source, and if the sources are the same (e.g., same file or same code stream), it assigns the “class” code source all or some of the permissions associated with the “matching” code sources. Specifically, Gong does not disclose a “*security policy specification*”, nor does it disclose a “*security policy specification [that] defines multiple policy levels*”. (Emphasis added).

Accordingly, Gong fails to disclose “generating a permission grant set... *if application evidence for the at least one application satisfies at least one condition specified in a security policy specification for trusting the application, wherein the security policy specification defines multiple policy levels,*” as recited in Claims 6, 22, and 33, and the similar features recited in Claim 44. (Emphasis added).

For these reasons, Claims 1, 6, 17, 22, 33, and 44 are allowable over Gong.

Claims 2 - 5, 7 - 16, 18 - 21, 23 - 35, 37 - 43, and 45 - 48

Claims 2 - 5, 7 - 16, 18 - 21, 23 - 35, 37 - 43, and 45 - 48 depend from independent Claims 1, 6, 17, 22, 33, and 44 and are allowable at least due to their dependency from Claims 1, 8 and 16, as well as for the features that they recite.

REJECTIONS UNDER 35 U.S.C. §103

The Office rejected Claims 3, 12, 19, 28, 34, and 40 under 35 U.S.C. §103(a) as being unpatentable over Gong in view of Lao et al. U.S. Patent Application No. 2003/0220880 A1 (“Lao”). Applicant respectfully traverses this rejection.

Lao U.S. Patent Application No. 2003/0220880

Lao discloses a method and computer system for licensing network services. The method includes determining the rights expression information associated with the distributed network service, the rights expression information indicating a manner of use of the distributed network service. The method then controls consumption of the distributed network service based on the rights expression information. (Paragraph 0008).

Regarding Claims 1 and 17

As discussed above, Gong fails to teach or suggest “receiving a manifest defining a plurality of code assemblies that are members of at least one application, *wherein the manifest defines at least one trusted application and application evidence for making a trust decision,*” as recited in Claims 1 and 17.

(Emphasis added). Lao fails to cure the deficiency of Gong.

Regarding Claims 6, 22, 33, and 44

Also as discussed above, Gong fails to teach or suggest “generating a permission grant set... *if application evidence for the at least one application satisfies at least one condition specified in a security policy specification for trusting the application, wherein the security policy specification defines multiple policy levels,*” as recited in Claims 6, 22, and 33, and the similar features recited in Claim 44. (Emphasis added). Lao fails to cure the deficiency of Gong.

Thus, Gong and Lao, whether taken alone or in combination (assuming for the sake of argument that they can be combined), fails to disclose or suggest all the features of Claims 1, 6, 17, 22, 33, and 44. Claims 3, 12, 19, 28, 34, and 40 depend from independent Claims 1, 6, 17, 22, 33, and 44 and are allowable by virtue of their dependency, as well as for additional features that they recite.

CONCLUSION

Applicant respectfully submits that Claims 1-35 and 37-48 are in condition for allowance. Applicant respectfully requests reconsideration and issuance of the subject application. Should any matter remain unresolved, the undersigned respectfully requests a telephone conference with the Examiner to resolve any outstanding matter.

Respectfully Submitted,

Date: 28 January 2008

By: 

Jeffrey Valley
Lee & Hayes, PLLC
Reg. No. 50,563
(509) 324-9256 x262